

REMARKS

The above amendments to the above-captioned application along with the following remarks are being submitted as a full and complete response to the Office Action dated August 22, 2005. In view of the above amendments and the following remarks, the Examiner is respectfully requested to give due reconsideration to this application, to indicate the allowability of the claims, and to pass this case to issue.

Status of the Claims

Claims 15-17 are under consideration in this application. Claims 1-14 are being cancelled without prejudice or disclaimer. Claim 15 is being amended, as set forth in the above marked-up presentation of the claim amendments, in order to more particularly define and distinctly claim applicants' invention. New claims 16-17 are being added to recite other embodiments described in the specification.

All the amendments to the claims are supported by the specification. Applicants hereby submit that no new matter is being introduced into the application through the submission of this response.

Prior Art Rejections

Claims 1, 2 and 5-15 were rejected under 35 U.S.C. § 103(a) as being unpatentable over US Patent No. 6,247,041 to Krueger et al. (hereinafter "Krueger") in view of US Patent No. 6,609,159 to Dukach et al. (hereinafter "Dukach") and further in view of US Patent No. 6,801,938 to Bookman et al. (hereinafter "Bookman"), and claims 3 and 4 under 35 U.S.C. § 103(a) on the grounds of being unpatentable over Krueger in view of Dukach and Bookman '938, and further in view of US Patent No. 6,424,988 to Lurndal (hereinafter "Lurndal"). These rejections have been carefully considered, but are most respectfully traversed.

The data transfer method of the invention (for example the embodiment depicted in Figs. 1, 4 & 9), as now recited in claim 15 for a data processing system (for example the embodiment depicted in Fig. 1; pp. 9-10, 20-23) which allows both processes to be executed in a first data processor 110 and in a second data processor 160 to communicate with each other by direct data transfer between user-spaces of the data processors (i.e., "*without the need for copying into the OS space*" p. 2, line 12; "*without any operation system involvement*" p. 1, line 10), wherein communication via a first virtual interface connection 10

(i.e., from VI 131 to VI 181) between a first process 112 in the first data processor 110 and a second process 162 in the second data processor 160 is taken over by a second virtual interface connection 10 (i.e., from VI 132 to VI 182) between a third process 115 in the first data processor 110 and the second process 162; the method comprising: setting a flag if the first process 112 receives data which is transmitted via the first virtual interface connection 10; issuing, from the first process 112 to the second process 162, a request 912 (Fig. 9) for intermission of data transmission to the first process 112; intermitting, in the second process 162, data transmission to the first process 112 and replying from the second process 162 to the first process 112 an acknowledge (e.g., “*report of completion of data transmission intermission 913*” p. 21, lines 1-4) of the request 912 for intermission of data transmission; after the acknowledge 913 of the request 912 is received in the first process 112; copying the data which is received in the first process 112 to the third process 115 (“The parent process receives the report 913 at time D and, at time Y after that, makes a copy 922 of received data into the child process” p. 21, lines 5-7); establishing the second virtual interface connection 20 between the third process 115 and the second process 162 in response to a reconnection request 914 (Fig. 9) issued from the first process 112 ; and taking over the communication from the first virtual interface connection 10 by the second virtual interface connection 20 to continue the communication.

The prior art VIA (Virtual Interface Architecture), which enables direct data transfer between user-spaces without the need for copying into the OS space, was depicted in Fig. 3 and described on page 2, 1st full paragraph and pages 3-4 of the specification. “*There is a problem that a child process created by the fork cannot communicate with the client because it cannot establish VI connection with the client* (p. 3, last paragraph).” By providing addition VI in each processor, the invention can establish a second virtual interface connection 20 (between a third process 115 in the first data processor 110 and the second process 162 in the second processor 160) to take over the communication from the first virtual interface connection 10 (between a first process 112 in the first data processor 110 and a second process 162 in the second data processor 160) in response to a request from the first process 112 to continue the communication.

Applicants contend that none of the cited prior art references teaches or suggests “after the acknowledge 913 of the request 912 for intermission of data transmission is received in the first process 112, copying the data which is received in the first process 112 to the third process 115 and taking over the communication from the first virtual interface

connection 10 by the second virtual interface connection 20 to continue the communication“ according to the invention.

As admitted by the Examiner (p. 3, last paragraph of the outstanding Office Action), Krueger fails to teach any virtual interfaces, and a third step in which the second connection takes over the communication from the first connection in response to a request from the first process to continue the communication.

Dukach was relied upon by the Examiner to teach the third step. However, Dukach only have a second connection takes over the communication from the first connection via its own socket. The socket as shown in Fig. 10 is totally within the OS space 138. Dukach has nothing to do with Virtual Interface Architecture (VIA) such that there in no VIA module, or VI, or any virtual interface connection at all. Dukach simply does not establish a second virtual interface connection 20 (between a third process n the first data processor and the second process in the second processor) to take over the communication from the first virtual interface connection 10 (between a first process in the first data processor and a second process in the second data processor) in response to a request from the first process to continue the communication as the invention. Moreover,

Bookman was relied upon by the Examiner to compensate for the deficiencies of Krueger and Dukach. However, contrary to the Examiner’s assertion (p. 3 last 2 lines; p. 4, lines 1-3 of the outstanding Office Action) that Bookman’s player process that has an established virtual port connection taking over communication from the leader process to form and continue communication via its own port, Bookman’s player process is spawned (may be later terminated) by a corresponding section leader process (col. 11, line 60-61; Fig. 6), rather than taking over any operation of the corresponding section leader process. In particular, if a player process 605-610 failure occurs, section leader processes 603-604 help program operation, terminating their controlled player processes and notifying other section leaders to do the same (col. 12, lines 6-10).

In addition, Bookman teaches away from the invention by transmitting data *in parallel* (co-existing) between plural player process and plural downstream player processes (Fig. 7; col. 12, lines 47-59; Abstract), rather than having one virtual interface connection *taking over* another virtual interface connection as the invention. It is well established that a rejection based on cited references having principles that teach away from the invention is improper.

Dukach and Lurndal was relied upon by the Examiner to teach claims 3-4. Dukach was relied upon by the Examiner to teach the copying step of the invention (p. 7, paragraph b. of the outstanding Office Action). However, Dukach merely copies all of variables, data structure and file descriptors to pipes and network communications from a parent process to a child process whenever the child process is created (col. 9, line 63 to col. 10, line 3), rather than “after the acknowledge 913 of the request 912 for intermission of data transmission is received in the first process 112”. Dukach simply fails to disclose the data copy timing from a parent process to a child process of the invention.

Lurndal’s first process (process manager PM) issues a slot request requesting for allocating a slot of distributed process directory and second process (process directory port group PDPG 405), the PDPG replies, and then, the PM assigns the designated slot to newly created child process (col. 11, line 66 to col. 12, line 53). Those steps constitute a request-respond correspondence between processes for assuring procedure integrity. Lurndal is silent regarding Virtual Interface Architecture (VIA) such that there is no VIA module, or VI, or any virtual interface connection at all. Lurndal simply does not establish a second virtual interface connection 20 (between a third process n the first data processor and the second process in the second processor) to take over the communication from the first virtual interface connection 10 (between a first process in the first data processor and a second process in the second data processor) in response to a request from the first process to continue the communication as the invention. Further more, Lurndal also fails to disclose the data copy timing from a parent process to a child process of the invention..

Applicants contend that the cited references or their combinations fail to teach or disclose each and every feature of the present invention as disclosed in the independent claim 15. As such, the present invention as now claimed is distinguishable and thereby allowable over the rejections raised in the Office Action. The withdrawal of the outstanding prior art rejections is in order, and is respectfully solicited.

Conclusion

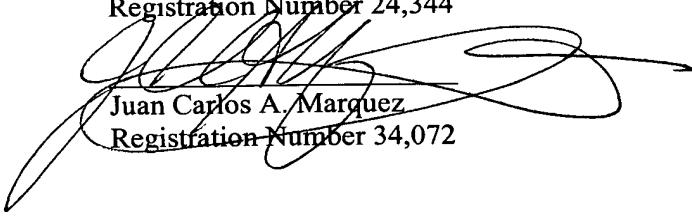
In view of all the above, clear and distinct differences as discussed exist between the present invention as now claimed and the prior art reference upon which the rejections in the Office Action rely, Applicants respectfully contend that the prior art references cannot anticipate the present invention or render the present invention obvious. Rather, the present invention as a whole is distinguishable, and thereby allowable over the prior art.

Favorable reconsideration of this application is respectfully solicited. Should there be any outstanding issues requiring discussion that would further the prosecution and allowance of the above-captioned application, the Examiner is invited to contact the Applicants' undersigned representative at the address and phone number indicated below.

Respectfully submitted,

Stanley P. Fisher
Registration Number 24,344

Juan Carlos A. Marquez
Registration Number 34,072



REED SMITH LLP
3110 Fairview Park Drive, Suite 1400
Falls Church, Virginia 22042
(703) 641-4200

February 21, 2006

SPF/JCM/JT